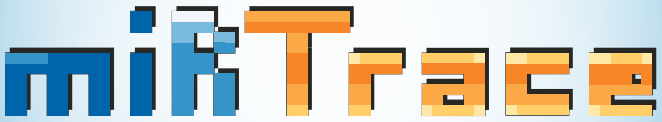


About

miRTrace: a tool for quality control and tracing taxonomic origins of microRNA sequencing data

miRTrace is a quality control and taxonomic tracing tool developed specifically for small RNA sequencing data (sRNA-Seq). miRTrace has two modes:

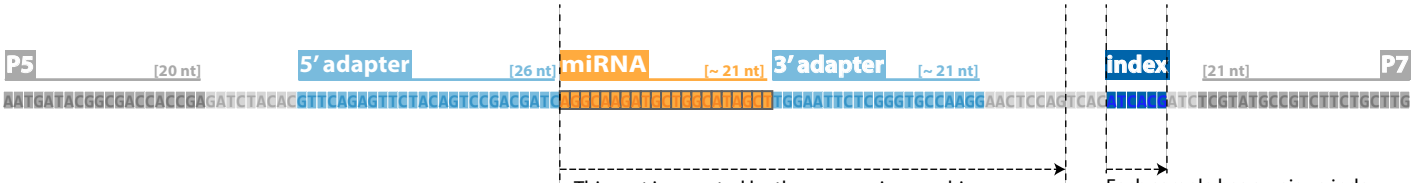
- QC mode is applicable for quality control of sRNA-Seq
- Trace mode is applicable for tracing taxonomic origins of sRNA-Seq

In the QC mode, each sample is characterized by profiling sequencing quality, read length, sequencing depth and miRNA complexity, the amounts of miRNAs versus undesirable sequences (derived from tRNAs, rRNAs and sequencing artifacts) and the composition of clade-specific miRNAs.

In the Trace mode, miRTrace can accurately resolve taxonomic origins of small RNA-Seq data based on the composition of clade-specific miRNAs. This feature can be used to detect cross-clade contaminations. It can also be applied for more specific applications in forensics, food safety and clinical diagnosis, such as tracing origins of food or detecting parasite microRNAs in host samples.

Workflow

Illumina TruSeq Small RNA library read structure



This part is reported by the sequencing machine as raw reads in the FASTQ file. The read length depends on how many cycles you defined for sequencing.

Each sample has a unique index. If you pooled several samples together, this part is used for sample demultiplexing.

Types of reads in a sRNA-Seq library

Here is an example to show potential types of reads in a normal sRNA-Seq data. The insert sequences that are located between the 5' and 3' adapters could be miRNA, other types of small RNA, degraded RNA derived from tRNA, rRNA, artifact sequences (see reference database section) or even empty (dimers of adapters or PCR primers).

miRTrace workflow

QC process

Remove low-quality reads

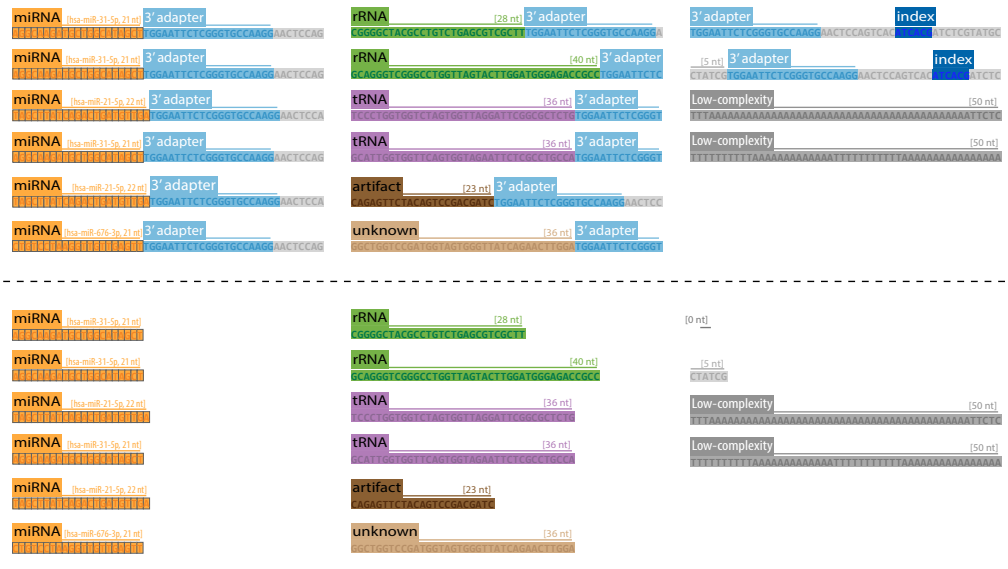
Remove the reads if more than 50% of nucleotides have a PHRED score less than 20.

Trim 3' adapters

Reads are processed to search for the first 8-mer of the 3' adapter, if match found, the last appearing match and together with the subsequent nucleotides are trimmed. If no match is found, the 3' end of read is aligned to the first 7-mer of 3' adapter, where the largest match is trimmed.

Remove low-complexity read

Remove the reads that have highly repeated nucleotides. Remove the reads containing any ambiguous nucleotide, e.g. N.



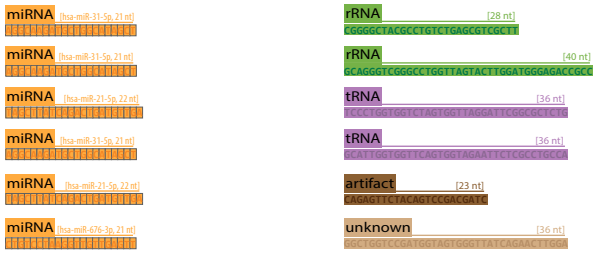
Remove short reads (< 18nt)

Remove short reads that are unlikely to be unambiguously traced back to the reference sequences.

Figure 1 Matrix 1 **Phred score**

Figure 2 Matrix 2 **Length distribution**

Figure 3 Matrix 3 **QC statistics**



FASTA With the option "--write-fasta", the reads that passed the QC process are recorded in one FASTA file per sample, which can be found in the output folder "qc_passed_reads.all.collapsed". The FASTA files can be used as input for downstream analysis, like SeqBuster (miRNA expression analysis) or miRDeep (*de novo* miRNA identification).

RNA type

The QC qualified reads are mapped to the reference rRNA, tRNA, miRNA hairpin and artifact sequences. The mapped reads are counted and shown as fractions in the figure and raw counts in the matrix.

Matrix 4 Figure 4

FASTA With the option "--write-fasta", the unknown reads are recorded in the output folder "qc_passed_reads.rnatype_unknown.collapsed". To figure out what these unknown reads are, we suggest to blast the top 10 most abundant sequences to the NCBI nucleotide correction (nr/nt).

miRNA complexity

The QC qualified reads are mapped to the miRNA hairpin sequences. The distinct miRNA hairpins that have reads aligned to them are counted and shown as accumulated distinct miRNA genes in the matrix. The more distinct miRNA hairpins detected the more complex the sample is.

Matrix 5 Figure 5

Clade-specific miRNA detection

The QC qualified reads are mapped to the catalog of clade-specific miRNA sequences that are curated from miRBase v21 based on the clade-specific miRNA family numbers. The mapped reads are summed up by clade and unique sequences, shown in matrix 6 and 7 respectively.

Matrix 6 Matrix 7 Figure 6

miRTrace has two modes:

- In **QC mode**, miRTrace performs all above modules.
- In **Trace mode**, miRTrace performs "QC process" and "Clade-specific miRNA detection".

Reference databases (can be found in miRTrace package)

1. Ribosomal RNA (rRNA) sequences (reference_rRNAs.fa)
   Ribosomal RNA sequences are curated from NCBI Nucleotide (https://www.ncbi.nlm.nih.gov/nuccore), Silva (https://www.arb-silva.de/) and Ensembl database 2015. The database is in FASTA format and contains 5,045 sequences. It covers 160 miRBase species and four types of rRNAs: 28S, 18S, 5.8S and 5S. 23 out of 160 species have rRNA sequences of all four types of rRNAs. An exception is *Drosophila* (fruit fly), which has the specific 2S rRNA. So we also included the 2S rRNA sequence (GenBank: U20145.1) to the database.

2. Transfer RNA (tRNA) sequences (reference_tRNAs.fa)
   Transfer RNA sequences are curated from tRNAdb (http://trna.bioinf.uni-leipzig.de/DataOutput/) and mitotRNAdb (http://mttrna.bioinf.uni-leipzig.de/mtDataOutput/) 2015. The database is in FASTA format and contains 3,599 sequences. It covers 78 miRBase species.

3. miRNA hairpin sequences (miRNA_hairpin_v21.fa)
   miRNA hairpin sequences (v21) that used by miRTrace can be download from miRBase (http://www.mirbase.org/ftp.shtml).

4. Artifact sequences (artifact_sequences.fa)
   The artifact sequences in FASTA format are curated from the Illumina adapter sequence document. (https://support.illumina.com/downloads/illumina-customer-sequence-letter.html).

5. Clade-specific miRNA family numbers (clade-specific_miRNA_families.txt)
   The clade-specific miRNA families are curated based on [1-5]. The list comprises clade-specific miRNA families from 14 clades: primates, rodents, birds and reptiles, fish, echinoderms, lophotrochozoa, insects, nematodes, sponges, dicots, monocots, gymnosperms, lycopods, bryophytes. The first column shows clade names. The second column shows the corresponding clade-specific miRNA family numbers separated by spaces. For example "primates 1200" would mean that the miRNA family 1200 is exclusively expressed in the primate clade. Based on miRBase v21, the family 1200 has two annotated entries: the human miRNA hsa-miR-1200 and the *Pongo pygmaeus* miRNA ppy-miR-1200.

References:
[1] Peterson, K. J., Dietrich, M. R., & McPeek, M. A. (2009). MicroRNAs and metazoan macroevolution: insights into canalization, complexity, and the Cambrian explosion. Bioessays, 31(7), 736-747.
[2] Sempere LF, Cole CN, McPeek MA, Peterson KJ. 2006. The phylogenetic distribution of metazoan microRNAs: insights into evolutionary complexity and constraint. J. Exp. Zool., 306B: 575–588. doi:10.1002/jez.b.21118
[3] Wheeler BM, Heimberg AM, Moy VN, Sperling EA, Holstein TW, Heber S, Peterson KJ. 2009. The deep evolution of metazoan microRNAs. Evol Dev 11: 50-68.Evol 306: 575-588.
[4] Lyson, TR, Sperling, EA, Heimberg, AM, Gauthier, YES, King, BL, & Peterson, KJ (2012). MicroRNAs support a turtle + lizard clade. Biology letters , 8 (1), 104-107.
[5] Taylor RS, Tarver JE, Hiscock SJ, Donoghue PC. 2014. Evolutionary history of plant microRNAs. Trends Plant Sci 19: 175-182.

Outputs

9 example samples

To illustrate what a miRTrace report looks like for good and poor-quality small RNA-Seq data sets, nine in-house libraries were prepared with considering common issues of library preparation (see below). To show the effect of adapter removal, one of the in-house data set was processed using an incorrect adapter sequence. All samples were subsampled to the same sequencing depth of around 3.5 million reads. Some of the in-house libraries are publicly available with GEO accession number: GSE118437.

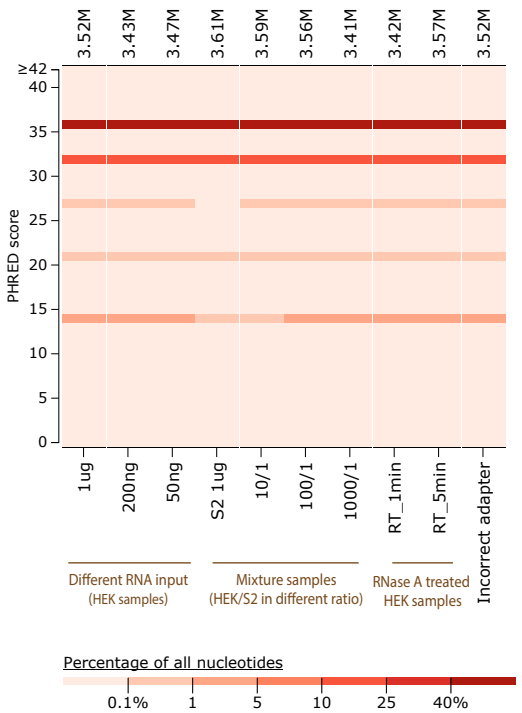
Different amount of RNAs for sRNA-Seq library preparation	Mixture samples (Human HEK and fruit fly S2 RNA mixed in different ratio)	RNase A treated samples	Incorrect adapter sequence
HEK293T RNA input 1 µg (control)	S2 RNA 1 µg (control)	RNase A for 1 min, room temperature	HEK293T sample was processed using an incorrect adapter sequence
HEK293T RNA input 200 ng	Mixed HEK and S2 RNA in ratio 10:1	RNase A for 5 min, room temperature	
HEK293T RNA input 50 ng	Mixed HEK and S2 RNA in ratio 100:1		
	Mixed HEK and S2 RNA in ratio 1000:1		
Purpose: To mimic samples with low-quantity of RNAs, e.g. clinical or body fluid samples. Usually 1 ug RNA is recommend for a standard Illumina library preparation.	Purpose: To mimic samples that are contaminated by RNAs from another species.	Purpose: To mimic degraded samples, which usually have low RIN value.	

6 output figures

In the QC mode, there are 6 output figures in the QC HTML report. Summary table is available in the end of the webpage.
In the Trace mode, there is 1 output figure: Figure 6 "clade-specific miRNA profile" in the Trace HTML report.

The HTML report is highly interactive. For example you can click the warning mark to know cutoffs. Click or ctrl-click one or multiple samples to show the information (in the legend) for the selected samples. Use key shortcuts A and D to reselect the left and right sample. Use key shortcuts W and D to show the previous and next figure.

Figure 1 **PHRED Score Distribution**
Percentage of nucleotides with given PHRED score.



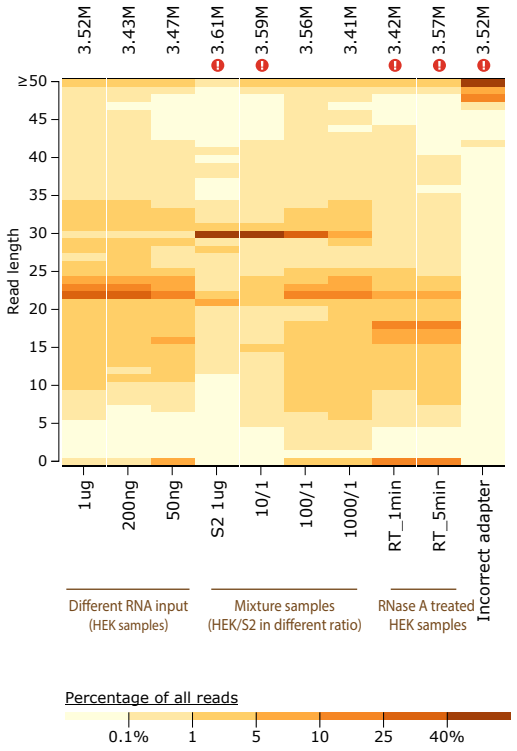
The plot shows how many nucleotides in each sample are likely to be correctly sequenced. Each bar represents a sample and the y-axis shows the Phred score. The numbers on top of bars represent the total number of reads.

Each nucleotide is assigned a Phred score (Q) by the sequencing machine. The higher the Phred score, the lower the likelihood is that the sequencing machine called the nucleotide wrong. For example, Q40 means an error possibility 0.0001 (1 in 10,000 nt is incorrectly called) and Q30 means an error possibility 0.001 (1 in 1,000).

The warning is given if > 50% of nucleotides have a PHRED score < 20.

Figure 2 **Read Length Distribution**

Percentage of reads of each length.



The plot shows the read length distribution after adapter removal. For standard small RNA-Seq data, we expect to see strong bands at around 22 nt, which is a typical length of animal miRNAs.

- The warning is given if < 25% of reads have a length between 20 and 25.

Demo sample comments

Different RNA input: the potential miRNA bands at around 22 nt are faded with decreasing the amount of input RNAs, indicating that the low-quantity samples have less amount of miRNA reads.

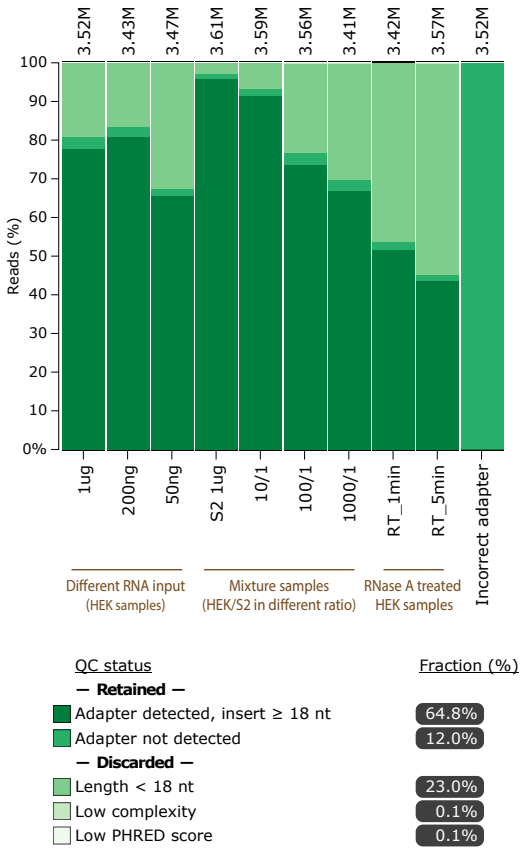
Mixture samples: the reads of 30 nt are 2S ribosomal RNAs, which are specific for *Drosophila*. As expected, the proportion of 2S rRNAs decreases when the ratio of HEK RNAs is increased.

RNase A treated samples have relatively short reads compared to the normal sample (HEK 1ug), indicating RNA degradation.

For the sample using an **incorrect adapter** as reference for adapter trimming, most reads are around 50 nt. Because the adapter sequences have not been trimmed.

Figure 3 **Quality Control Statistics**

Percentage of reads with given quality control status.



The plot shows statistics of the QC process (see work flow), including low PHRED score filtering, adapter removal, low complexity filtering and length filtering. The reads are categorized into five groups (see plot legend). The "Adapter removed, remaining seq ≥ 18nt" and "Adapter not detected" reads are retained for the analysis shown in figure 4-6. We call them "QC qualified" reads.

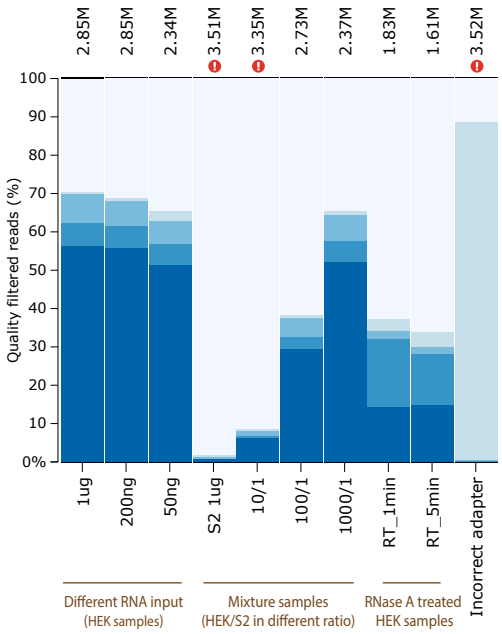
- The warning is given if < 25% of reads are retrained.

Demo sample comments

As expected, the sample processed with **incorrect adapter** has a lot of "adapter not detected" reads.

Figure 4 **RNA Type**

Percentage of reads of each RNA type.



RNA type	Fraction (%)	— Reference seqs — Detected	Total
miRNA	26.7%	1147	1881
rRNA	4.7%	48	48
tRNA	3.6%	417	431
Artifacts	12.6%	54	56
Unknown	52.5%	-	-

The plot shows the composition of detected RNA types in each sample. The numbers on top of bars are the QC qualified read counts. Please note that the RNA type detection highly depends on the coverage of reference databases, which varies from species to species.

- The warning is given if < 10% of reads are identified as miRNAs.

Demo sample comments

Different RNA input: the percentage of miRNA reads decreases slightly with decreasing the amount of input RNAs.

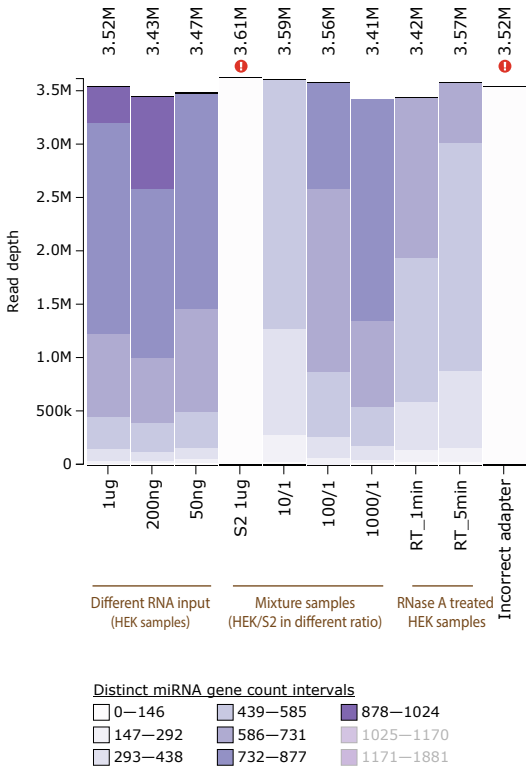
Mixture samples: it is not surprising to detect a small percentage of miRNAs in S2 1ug sample when human sequences are used as reference. The percentage of miRNAs increases dramatically with increasing ratio of HEK293T RNAs.

RNase A treated: samples have low percentage of miRNA reads compared to normal samples (e.g. the 1 ug sample), indicating RNA degradation.

For the sample processed with **incorrect adapter**, almost all reads are "artifacts", indicating the insert sequences have not been properly identified because the adapter was not trimmed.

Figure 5 **miRNA Complexity**

Number of detected distinct miRNA genes as function of read depth.



The plot shows two types of information: 1) the number of raw reads in each sample (represented by the height of the bar) before any processing and 2) the miRNA complexity. We assess complexity by the number of distinct miRNA genes that are observed as a function of sequencing depth. This is in essence a saturation analysis. The gradient colour changes from white to dark purple as more distinct miRNA hairpins have been detected.

- The warning is given if < 10% of the known miRNA genes in the given species have been detected.

Demo sample comments

Different RNA input: samples prepared with low RNA input and with standard input have comparable miRNA complexity.

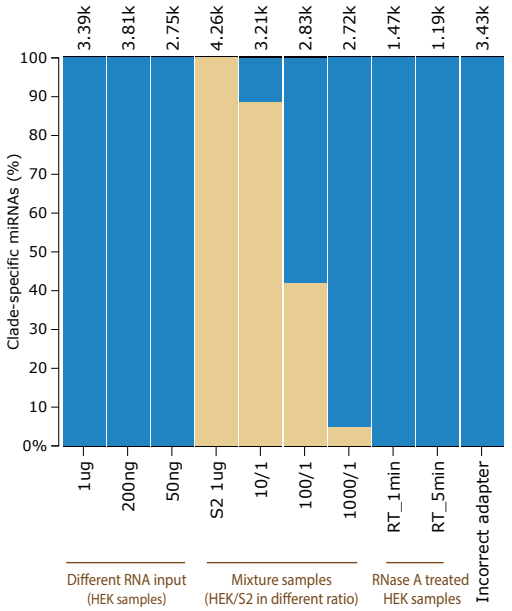
Mixture samples: the reference is human sequences. This explains why the drosophila sample (S2 1ug) has barely no miRNA genes detected. The detected miRNAs are likely conserved in both *drosophila* and human. The human miRNA complexity clearly increases with increasing ratio of HEK293T RNAs.

RNase A treated samples have lower miRNA complexity compared to the standard sample (HEK 1ug). This is consistent with the experimental design.

For the sample processed with **incorrect adapter**, almost no miRNA is detected. Because the adapter was not trimmed.

Figure 6 **Contamination (in QC mode) / Clade-specific miRNA profile (in Trace mode)**

Percentage of clade-specific miRNA-reads belonging to each clade.



Clade	Fraction (%)	— *miRNA families* — Detected	Total
Primates	71.0%	33	59
Rodents		0	16
Birds/Reptiles		0	21
Fish		0	7
Echinoderms		0	9
Lophotrochozoa		0	26
Insects	29.0%	19	61
Nematode		0	4
Sponges		0	8
Dicots		0	175
Monocots		0	62
Gymnosperms		0	6
Lycopods		0	33
Bryophytes		0	67

The plot shows the types and proportions of clade-specific miRNAs detected in each sample. For a good sRNA-Seq library, we expect that almost all of the clade-specific miRNA sequences are assigned to the expected clade. The plot allows us to check if there are potential cross-clade contaminations introduced during library preparation, demultiplexing or by sample mishandling.

Demo sample comments

Mixture samples: the proportion of primate specific miRNAs clearly increases with increasing ratio of HEK293T RNAs.

7 output matrices

In the QC mode, there are 7 output matrices (see below).
In the Trace mode, there are 2 output matrices: Matrix 6 and 7.

The 7 miRTrace output matrices correspond to the output figures. The matrices are tab delimited. Matrix 7 is a detailed version of matrix 6. The matrix format is illustrated as follows. Matrix content corresponds to the example small RNA-Seq sequences from the "workflow" section. Please note that the matrices may have omitted rows for brevity reasons.

Matrix 1 **PHRED Score Distribution**
(mirtrace-stats-phred.tsv)

PHRED_SCORE	Sample1 (nucleotide counts)	Sample2 (nucleotide counts)	...
0	count		
1	count		
2	count		
3	count		
4	count		
5	count		
⋮	⋮	⋮	
up to 42			

Matrix 2 **Read Length Distribution**
(mirtrace-stats-length.tsv)

LENGTH	Sample1 (read counts)	Sample2 (read counts)	...
0	1		
5	1		
21	4		
22	2		
23	1		
28	1		
36	3		
40	1		
⋮			
up to 50			

Matrix 3 **Quality Control Statistics**
(mirtrace-stats-qcstatus.tsv)

QC_STATUS	Sample1 (read counts)	Sample2 (read counts)	...
LOW_PHRED	2		
LOW_COMLPEXITY	2		
ADAPTER_NOT_DETECTED	0		
ADAPTER_REMOVED_LENGTH_OK	12		
LENGTH_SHORTER_THAN_18	2		

Matrix 4 **RNA Type**
(mirtrace-stats-rnatype.tsv)

RNA_TYPE	Sample1	Sample2	...
miRNA	6		
rRNA	2		
tRNA	2		
artifact	1		
unknown	1		

Matrix 5 **miRNA Complexity**
(mirtrace-stats-mirna-complexity.tsv)

DISTINCT MIRNA HAIRPINS	Sample1 (accumulated read counts)	Sample1 (accumulated read counts)	...
1	1		
2	3		
3	6		
⋮	⋮	⋮	

Matrix 6 **Cross-clade miRNA clade counts**
(mirtrace-stats-contamination_basic.tsv)

CLADE	Sample1 (sum of clade-specific miRNA reads)	Sample2 (sum of clade-specific miRNA reads)	...
primates	0		
rodents	0		
insects	0		
nematodes	0		
dicots	0		
⋮	⋮		
up to 14 clades			

Matrix 7 **Clade-clade miRNA sequence counts**
(mirtrace-stats-contamination_detailed.tsv)

CLADE	FAMILY_ID	MIRBASE_IDS	SEQ	Sample1 (counts)	Sample2 (counts)	...
primates	584	hsa-miR-584; ptr-miR-584; mml-miR_584	TTATGGTTTGCCTGGGACTG			
primates	584	hsa-miR-584	TCAGTTCCAGGCCAACCAGG			
rodents	⋮	⋮	⋮	⋮	⋮	
insects						
⋮	⋮	⋮	⋮	⋮	⋮	
up to 14 clades						

The first column shows clade names. The second column represents clade-specific family numbers. The third column shows the miRBase entries that belong to the family in column two and are identical in their frist 20 nts. The fourth column shows the identical sequences in 20 nts. The remaining columns show the number of reads that the first 20 nts match with the sequences in column four.

2 output FASTA files per sample when using the option "--write-fasta"

In the QC mode, there are 2 output FASTA files ① and ② per sample.
In the Trace mode, there are 1 output FASTA file ① per sample.

① FASTA files in output folder "qc_passed_reads.all.collapsed"

Reads that pass the quality control (QC) are deposited in a FASTA file per sample. Identical reads are collapsed into a single entry, unless the option "--uncollapse-fasta" is used. Note that the read abundance is augmented to the sequence ID as "_xNNNN" where "NNNN" is the read abundance. In the example below, the number 1000 after the "x" is the read abundance. The entries of FASTA file are sorted by read abundance.

```
>seq_0_x1000
TGAGGTAGTAGTTTGTGCT
```

② FASTA files in output folder "qc_passed_reads.rnatype_unknown.collapsed"

Unknown reads as defined in the "RNA Type" figure are deposited to this FASTA file. The unknown reads could be other types of small RNAs that are not covered by miRTrace, artifacts or degraded RNAs etc. To figure out what these unknown reads are, we suggest to blast the top 10 most abundant unknown reads to the NCBI nucleotide database (nr/nt, https://blast.ncbi.nlm.nih.gov/Blast.cgi).

Manual

REQUIRED

* Java 1.7 or greater. (I.e. Linux, Mac OS X, Windows etc. are supported.)

* An absolute minimum of 1 GB of RAM is required. Significantly more may be needed to process large samples effciently. We recommend a at least 4 GB of RAM, preferrably 16 GB. For compute-nodes with many cores, more RAM may be needed for optimal performance.

* For the HTML report, any "modern" web browser that supports the D3.js library. This includes Chrome, Firefox, Internet Explorer 9+, Opera and Safari. Note that reports with a very large number of samples (50-100 or more) may be slow or inaccessible on typical hardware.

*To build miRTrace, additional software is required, such as Python 3.

JAVA MEMORY ISSUE AND WRAPPER SCRIPT

Java programs run inside a so called virtual machine where the maximum amount of available memory (RAM) is determined when the program starts. The default value varies across systems and can be lower than ideal or too low to run miRTrace at all.

To handle this problem (for any Java software), the user can always set a manual memory allocation for the Java Virtual Machine (JVM). It's done using the -Xms and -Xmx flags, e.g.:
"java -jar -Xms2G -Xmx2G mirtrace.jar".

In order to make miRTrace more user friendly, we've added a wrapper script (also known as a start script) that automatically determines a suitable memory allocation and sends these parameters to the JVM. It's used by running "./mirtrace" followed by the command line arguments outlined elsewhere in this manual. If this script has not been made executable during the installation process this has to be done manually. To make it runnable by the user and group, type: "chmod ug+x mirtrace". [Note the wrapper script is not available on Windows systems]

This script usually works but may fail under e.g. these conditions:
1) The user is running miRTrace on a Windows system.
2) Python or some necessary library is unavailable or of an incompatible version on the system.
3) The 'java' command points to a JVM of the wrong version so that a full path to the Java executable has to be used (the script can be easily changed in this case).
4) Unforeseen format of system tool output or file system locations in a particular installation.
5) Very large FASTQ files (in relation to the total system RAM) may require more than 50% of the total system RAM to process.
6) Memory usage on the system is so high that miRTrace causes problems for other software running on the system.

If the wrapper script fails don't worry, just use the native JVM invocation syntax, like "java -jar -Xms2G -Xmx2G mirtrace.jar".

COMMAND LINE USAGE

USAGE: **java -Xms<mem in MB>M -Xmx<mem in MB>M -jar <MIRTRACE JAR> MODE [-s SPECIES] [-a ADAPTER] [OTHER OPTIONS]... [FASTQ filenames]...**

NOTE: Please allocate a large amount of RAM to the Java JVM using the JVM parameters "-Xms" and "-Xmx". For e.g. an 8 GB RAM machine, the following is recommended: "-Xms4G -Xmx4G".

SIMPLE USAGE EXAMPLE (QC mode):
java -Xms4G -Xmx4G -jar mirtrace.jar qc --species hsa --adapter TGGAATTCT sample_A.fq sample_B.fq.gz

SIMPLE USAGE EXAMPLE (Trace mode):
java -Xms4G -Xmx4G -jar mirtrace.jar trace --adapter TGGAATTCT sample_A.fq sample_B.fq.gz

MODES
The first argument must specifiy what mode miRTrace should operate in. Available modes:
trace Trace mode. Produces a clade-specific miRNA profile report. --species is ignored.
qc Quality control mode (full set of reports). --species must be given.

ARGUMENT REQUIRED IN QC MODE:
-s, --species Species (miRBase encoding). EXAMPLE: "hsa" for ***H****ome* ***sa****piens.*
To list all codes, run "miRTrace --list-species".

SPECIFYING INPUT FILES USING A CONFIG FILE:
If the input samples are not given as arguments directly, a config file must be used.
-c, --config List of FASTQ files to process. This is a CSV (comma separated value) file, i.e. with one entry per row.

Each row consists of the following columns (only the first is required):
filename,name-displayed-in-report,adapter,PHRED-ASCII-offset

NOTE: the PHRED ASCII offset can typically be reliably auto-detected and is not necessary to specifiy.

EXAMPLE CONFIG FILE:
path/sample1.fastq,sample 1 (control),TGGAATTC
path/sample2.fastq,sample 2 (+drug X),TGGAATTC

OPTIONAL ARGUMENTS:
-a, --adapter <DNA sequence>. [DEFAULT: none]
-p, --protocol One of the following (read structure schematic in parens):
illumina (miRNA--3'-adapter--index) [DEFAULT]
qiaseq (miRNA--3'-adapter--UMI--3'-adapter--index)
NOTE: DO NOT SPECIFY AN ADAPTER WITH -p qiaseq. IT WILL BE IGNORED.
cats (NNN--miRNA--poly-A--3'-adapter--index)
nextflex (NNNN--miRNA--NNNN--3'-adapter--index)
-o, --output-dir Directory for output files. [DEFAULT: <file listing>.output]
-f, --force Overwrite output directory if it exists.
--enable-pipes Enable support for named pipes (fifos) as input. NOTE: Requires a config file with PHRED and adapter given for each entry. Input must not be compressed.
-w, --write-fasta Write QC-passed reads and unknown reads (as defined in the RNA type plot) to the output folder. Identical reads are collapsed. Entries are sorted by abundance.

OPTIONAL ARGUMENTS [FASTA OUTPUT] (Only relevant if --write-fasta given):
--uncollapse-fasta Write one FASTA entry per original FASTQ entry.

OPTIONAL ARGUMENTS [HTML REPORT OPTIONS]:
--title Set the report title.
--comment Add a comment to the generated report. Multiple arguments can be given.

OPTIONAL ARGUMENTS [PERFORMANCE / TROUBLESHOOTING]:
-t, --num-threads Maximum number of processing threads to use. [DEFAULT: <number of virtual cores>]
--verbosity-level Level of detail for debug messages. [default: 1]
--global-mem-reserve Amount of Java memory reserved for "housekeeping" tasks (in MB). Increase only if OutOfMemoryErrors are occuring. Decrease only if available system memory is very low. [Current value: 400 MB]

OPTIONAL ARGUMENTS [CUSTOM DATABASES]:
--custom-db-folder Folder containing user-generated reference databases.

HELP
--list-species List all available species and their codes.
--cite Show information about how to cite our paper.
-h, --help Display this help text.
-v, --version Display miRTrace version number.

How to get started?

INSTALLATION

1. Download the miRTrace package from (https://friedlanderlab.org/software/mirtrace/). The package contains mirtrace.jar (java program) and mirtrace (wrapper script) in the root directory and many subfolders.

2. Unzip the miRTrace package.

3. Open terminal, go to the miRTrace folder.

4. Check the java version by typing "java -version". miRTrace requires java version at least 1.7.

5. The tutorial examples about running miRTrace are available in the next few pages.

Potential installation problems and solutions

1. Incorrect JAVA version. miRTrace requires JAVA version 1.7 or higher.

This may be the issue for Mac computer, since Mac has java 1.6 as default version. If java 1.6 is applied to run miRTrace, you will get the error message "Exception in thread "main" java- .lang.UnsupportedClassVersionError: se/scilifelab/mirtrace/MiRTrace : Unsupported major.minor version 51.0".

Solution:
We recommend using OpenJDK or any other open source-licensed JDK.

2. The mirtrace wrapper script is not executable.

Please type:

```
chmod ug+x mirtrace
```

Example usage cases

Case 1: One sample

Input FASTQ file

doc/manual/tutorials/sample1.fastq.gz

Species: *Homo sapiens* (miRBase species code: hsa)
3' adapter sequence: TGGAATTCTCGGGTGCCAAGG

QC mode
(is applicable for quality control of sRNA-Seq data)

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar qc --species hsa --adapter TGGAATTCTCGGGTGCCAAGG doc/manual/tutorials/sample1.fastq.gz
```

OR command line for wrapper script: (Note not available for Window system)

```
./mirtrace qc --species hsa --adapter TGGAATTCTCGGGTGCCAAGG doc/manual/tutorials/sample1.fastq.gz
```

Output files:
Default is the working directory. For example: ./mirtrace (with time stamp)

mirtrace-stats-phred.tsv (Matrix 1)
mirtrace-stats-length.tsv (Matrix 2)
mirtrace-stats-qcstatus.tsv (Matrix 3)
mirtrace-stats-rnatype.tsv (Matrix 4)
mirtrace-stats-mirna-complexity.tsv (Matrix 5)
mirtrace-stats-contamination_basic.tsv (Matrix 6)
mirtrace-stats-contamination_detailed.tsv (Matrix 7)
mirtrace-results.json (an undocumented but convenient data structure containing all information from the report)
mirtrace-report.html (the HTML report)

TIP: To get the QC passed reads and unknown reads written to FASTA files, please use the option `--write-fasta`. See below.

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar qc --species hsa --adapter TGGAATTCTCGGGTGCCAAGG doc/manual/tutorials/sample1.fastq.gz --write-fasta
```

OR command line for wrapper script: (Note not available for Window system)

```
./mirtrace qc --species hsa --adapter TGGAATTCTCGGGTGCCAAGG doc/manual/tutorials/sample1.fastq.gz --write-fasta
```

In addition to the above output files, you will get the FASTA files:
Default is the working directory. For example: ./mirtrace (with time stamp)

qc_passed_reads.all.collapsed (contains QC qualified reads of each sample)
qc_passed_reads.rnatype_unknown.collapsed (contains unknwon reads of each sample reported in RNA type plot)

Trace mode
(is applicable for clade-specific miRNA detection)

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar trace --adapter TGGAATTCTCGGGTGCCAAGG doc/manual/tutorials/sample1.fastq.gz
```

OR command line for wrapper script: (Note not available for Window system)

```
./mirtrace trace --adapter TGGAATTCTCGGGTGCCAAGG doc/manual/tutorials/sample1.fastq.gz
```

Output files:
Default is the working directory. For example: ./mirtrace (with time stamp)

mirtrace-stats-contamination_basic.tsv (Matrix 6)
mirtrace-stats-contamination_detailed.tsv (Matrix 7)
mirtrace-report.html (the HTML report)

TIP: To get the QC passed reads and unknown reads written to FASTA files, please use the option `--write-fasta`. See below.

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar trace --adapter TGGAATTCTCGGGTGCCAAGG doc/manual/tutorials/sample1.fastq.gz --write-fasta
```

OR command line for wrapper script: (Note not available for Window system)

```
./mirtrace trace --adapter TGGAATTCTCGGGTGCCAAGG doc/manual/tutorials/sample1.fastq.gz --write-fasta
```

In addition to the above output files, you will get the FASTA files:
Default is the working directory. For example: ./mirtrace (with time stamp)

qc_passed_reads.all.collapsed (contains QC qualified reads of each sample)

Case 2: Multiple samples with identical 3' adapter

Input FASTQ files

doc/manual/tutorials/sample1.fastq.gz
doc/manual/tutorials/sample2.fastq.gz

Species: *Homo sapiens* (miRBase species code: hsa)
3' adapter sequence: TGGAATTCTCGGGTGCCAAGG

QC mode
(is applicable for quality control of sRNA-Seq data)

STEP 1: create a comma delimited config file. Each line represents one sample.

doc/manual/tutorials/config4case2

```
doc/manual/tutorials/sample1.fastq.gz,Sample1
doc/manual/tutorials/sample2.fastq.gz,Sample2
```

STEP 2: run the program.

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar qc --species hsa --adapter TGGAATTCTCGGGTGCCAAGG --config doc/manual/tutorials/config4case2
```

OR command line for wrapper script: (Note not available for Window system, please use the command line for mirtrace.jar)

```
./mirtrace qc --species hsa --adapter TGGAATTCTCGGGTGCCAAGG --config doc/manual/tutorials/config4case2
```

NOTE: If config file is applied, the default output will be a subdirectory to the directory containing the config file, e.g. doc/manual/tutorials/config4case2.output. An example of output content can be found in **Case 1** or "doc/manual/tutorials/tutorial_outputs/Case2_outputs/mirtrace.qc.outputs/"

TIP: If you don't want to use the config file, you can list the files you want to process as below.

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar qc --species hsa --adapter TGGAATTCTCGGGTGCCAAGG \
doc/manual/tutorials/sample1.fastq.gz doc/manual/tutorials/sample2.fastq.gz
```

OR command line for wrapper script: (Note not available for Window system, please use the command line for mirtrace.jar)

```
./mirtrace qc --species hsa --adapter TGGAATTCTCGGGTGCCAAGG doc/manual/tutorials/sample1.fastq.gz doc/manual/tutorials/sample2.fastq.gz
```

Trace mode
(is applicable for clade-specific miRNA detection)

STEP 1: create a comma delimited config file. (same as the QC mode, see above)

STEP 2: run the program.

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar trace --adapter TGGAATTCTCGGGTGCCAAGG --config doc/manual/tutorials/config4case2
```

OR command line for wrapper script: (Note not available for Window system, please use the command line for mirtrace.jar)

```
./mirtrace trace --adapter TGGAATTCTCGGGTGCCAAGG --config doc/manual/tutorials/config4case2
```

Case 3: Multiple samples with different 3' adapter

Input FASTQ files

doc/manual/tutorials/sample1.fastq.gz
doc/manual/tutorials/sample2.fastq.gz
doc/manual/tutorials/sample3.fastq.gz

Species: *Homo sapiens* (miRBase species code is hsa)

```
3' adapter TGGAATTCTCGG for sample1
           TGGAATTCTCGG for sample2
           ATCTCGTATGCC for sample3
```

QC mode
(is applicable for quality control of sRNA-Seq data)

STEP 1: create a comma delimited config file. Each line represents one sample.

doc/manual/tutorials/config4case3

```
doc/manual/tutorials/sample1.fastq.gz,Sample1,TGGAATTCTCGG
doc/manual/tutorials/sample2.fastq.gz,Sample2,TGGAATTCTCGG
doc/manual/tutorials/sample3.fastq.gz,Sample3,ATCTCGTATGCC
```

STEP 2: run the program.

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar qc --species hsa --config doc/manual/tutorials/config4case3
```

OR command line for wrapper script: (Note not available for Window system, please use the command line for mirtrace.jar)

```
./mirtrace qc --species hsa --config doc/manual/tutorials/config4case3
```

NOTE: If config file is applied, the default output will be a subdirectory to the directory containing the config file, e.g. doc/manual/tutorials/config4case3.output. An example of output content can be found in **Case 1** or "doc/manual/tutorials/tutorial_outputs/Case3_outputs/mirtrace.qc.outputs/".

Trace mode
(is applicable for clade-specific miRNA detection)

STEP 1: create a comma delimited config file. (same as the QC mode, see above)

STEP 2: run the program.

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar trace --config doc/manual/tutorials/config4case3
```

OR command line for wrapper script: (Note not available for Window system, please use the command line for mirtrace.jar)

```
./mirtrace trace --config doc/manual/tutorials/config4case3
```

Case 4: Using custom reference databases

For example, if the reference sequences for the species of interest are not available in miRTrace package.
Users can generate the custom reference databases for miRTrace.

Input FASTQ files	Custom databases
doc/manual/tutorials/sample4.fastq.gz	doc/manual/tutorials/crassostrea_gigas_rRNAs.fasta (downloaded from Silva) doc/manual/tutorials/crassostrea_gigas_miRNAs.fasta (downloaded from MirGeneDB) doc/manual/tutorials/crassostrea_gigas_tRNAs.fasta (empty) doc/manual/tutorials/crassostrea_gigas_artifacts.fasta (same as the default database)
Species: *Crassostrea gigas*, which is not available in miRBase v21 3' adapter TGGAATTCTCGG	

STEP 1: go to the subdirectory "scripts/custom_database_generation" in miRTrace bundle.

```
cd scripts/generate_custom_databases
```

STEP 2: generate custom reference databases that can be used by miRTrace using the script "generate-mirtrace-rnatype-database.py".
Note: to run the script, working directory must be the same dir as where the script resides. The script works with python 3.

```
python3 generate-mirtrace-rnatype-database.py --out-dir ../../doc/manual/tutorials/cgi_custom_databases \
--species-abbrev cgi --species-verbosename crassostrea_gigas \
--mirna-seqs ../../doc/manual/tutorials/crassostrea_gigas_miRNAs.fasta \
--rrna-seqs ../../doc/manual/tutorials/crassostrea_gigas_rRNAs.fasta \
--trna-seqs ../../doc/manual/tutorials/crassostrea_gigas_tRNAs.fasta \
--artifacts-seqs ../../doc/manual/tutorials/crassostrea_gigas_artifacts.fasta
```

STEP 3: go to the directory where the mirtrace.jar resides.

```
cd ../../
```

STEP 4: run the program with the option "--custom-db-folder" to let miRTrace use the custom databases instead of using the default databases.

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar qc --species cgi --custom-db-folder doc/manual/tutorials/cgi_custom_databases \
--adapter TGGAATTCTCGG doc/manual/tutorials/sample4.fastq.gz
```

OR command line for wrapper script: (Note not available for Window system, please use the command line for mirtrace.jar)

```
./mirtrace qc --species cgi --custom-db-folder doc/manual/tutorials/cgi_custom_databases --adapter TGGAATTCTCGG doc/manual/tutorials/sample4.fastq.gz
```

Case 5: analyzing sRNA-Seq data from non-Illumina protocols

miRTrace is applicable for the following protocols:

1. Illumina TruSeq Small RNA with read structure: miRNA - 3′ adapter - index [DEFAULT]
2. QiaSeq miRNA protocol with read structure: miRNA - 3′ adapter - UMI - 3′ adapter - index
3. NEXTflex Small RNA-Seq protocol with read structure: NNNN - miRNA - NNNN - 3′ adapter - index
4. CATS Small RNA-Seq protocol with read structure: NNN - miRNA - polyA - 3′ adapter - index

If you want to analyze small RNA-Seq data from non-Illumina protocols, e.g. QiaSeq, NEXTflex and CATS, please use the -p or --protocol option to specify protocol type. Here is an example to use miRTrace to analyze QiaSeq small RNA-Seq data.

Input FASTQ files
doc/manual/tutorials/QiaSeq_data.fastq.gz
Species: *Mus musculus* (miRBase species code: mmu) 3′ adapter sequence: AACTGTAG Protocol: QiaSeq miRNA

QC mode
(is applicable for quality control of sRNA-Seq data)

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar qc --species mmu --adapter AACTGTAG --protocol qiaseq doc/manual/tutorials/QiaSeq_data.fastq.gz
```

OR command line for wrapper script: (Note not available for Window system)

```
./mirtrace qc --species mmu --adapter AACTGTAG --protocol qiaseq doc/manual/tutorials/QiaSeq_data.fastq.gz
```

Trace mode
(is applicable for clade-specific miRNA detection)

Command line for mirtrace.jar:

```
java -jar -Xms4G -Xmx4G mirtrace.jar trace --adapter AACTGTAG --protocol qiaseq doc/manual/tutorials/QiaSeq_data.fastq.gz
```

OR command line for wrapper script: (Note not available for Window system)

```
./mirtrace trace --adapter AACTGTAG --protocol qiaseq doc/manual/tutorials/QiaSeq_data.fastq.gz
```